REPLACEMENT CLAIMS:

Please cancel claim 7.

1. **(Amended)** A method for implementing interrupts in a data processing system, comprising the steps of:

providing a first storage device having a plurality of inputs, each of the plurality of inputs being coupled by a respective physical conductor to one of a plurality of hardware-generated interrupt sources which selectively generate hardware interrupts and selectively storing the hardware interrupts, the first storage device providing one or more hardware-generated interrupt signals;

providing a second storage device having one or more inputs, each of the one or more inputs receiving and storing a predetermined one of a plurality of software-generated interrupt signals, at least some of the predetermined plurality of software-generated interrupt signals indicating an interrupt from a different source or of a different type than the hardware interrupts, the second storage device providing one or more software-generated interrupt signals;

coupling logic circuitry to the first storage device and the second storage device for receiving the one or more hardware-generated interrupt signals and the one or more software-generated signals, the logic circuitry providing an interrupt request signal which will cause an interrupt to occur in the data processing system; and

determining priority between two interrupts, a first interrupt being hardware-generated and a second interrupt being software-generated, when the two interrupts have a same prioritization level by choosing to service one of the hardware-generated first interrupt or the software-generated second interrupt.

2. (**Amended**) The method of claim 1 further comprising the step of:

assigning an interrupt prioritization level to storage locations of the first storage device and the second storage device.

4. (**Amended**) The method of claim 2 further comprising the step of:

assigning a portion of the plurality of software-generated interrupt signals stored in the second storage device to represent interrupts from some interrupt sources generating hardware interrupts and having an interrupt prioritization level which differs from the interrupt prioritization level of the plurality of hardware-generated interrupt sources coupled to the first storage device.

8. (**Amended**) A method for implementing interrupts in a data processing system, comprising the steps of:

providing a first storage device having a plurality of inputs, each of the plurality of inputs being coupled by a respective physical conductor to one of a plurality of hardware-generated interrupt sources which selectively generate hardware interrupts and selectively storing the hardware interrupts, the first storage device providing one or more hardware-generated interrupt signals;

providing a second storage device having one or more inputs, each of the one or more inputs receiving and storing a predetermined one of a plurality of software-generated interrupt signals, at least some of the predetermined plurality of software-generated interrupt signals indicating an interrupt from a different source or of a different type than the hardware interrupts, the second storage device providing one or more software-generated interrupt signals;

coupling logic circuitry to the first storage device and the second storage device for receiving the one or more hardware-generated interrupt signals and the one or more software-generated signals, the logic circuitry providing an

B1 Cont'd.

interrupt request signal which will cause an interrupt to occur in the data processing system; and

coupling enabling circuitry between the first and second storage devices and the logic circuitry, the enabling circuitry receiving the hardware-generated and software-generated interrupts and determining whether to pass the hardware-generated and software-generated interrupts to the logic circuitry for further processing.

9. (**Amended**) A data processing system with interrupt control circuitry, comprising:

a plurality of hardware interrupt sources;

a hardware interrupt storage device having a plurality of inputs, each of the plurality of inputs being coupled by an electrical conductor to one of a plurality of hardware interrupt sources, the hardware interrupt storage device storing hardware-generated interrupts and providing each of the hardware-generated interrupts at a predetermined output terminal;

a software interrupt storage device having a plurality of inputs, each of the plurality of inputs receiving a predetermined one of a plurality of software-generated interrupt signals, at least one of the software-generated interrupt signals corresponding to interrupt servicing of a portion of the data processing system which is not designated as a hardware interrupt source; and

logic circuitry coupled to the hardware interrupt storage device and the software interrupt storage device for providing a data processing system interrupt signal in response to receipt of either hardware-generated interrupts or software-generated interrupts, wherein the logic circuitry determines priority between two interrupts, a first interrupt being hardware-generated and a second interrupt being software-generated, when the two interrupts have a same prioritization level by choosing to service one of the hardware-generated first interrupt or the software-generated second interrupt.

17. **(New Claim)** The method of claim 2 further wherein the interrupt prioritization level is assigned to specific storage locations of the first and second storage device, the interrupt prioritization level of the plurality of hardware-generated interrupt sources coupled to the first storage device being permanently assigned, but assignment of the interrupt prioritization level of interrupt sources associated with the second storage device being variable by software control.

18. **(New Claim)** The method of claim 8 further comprising the step of:
assigning an interrupt prioritization level to storage locations of the first storage device and the second storage device.

19. **(New Claim)** The method of claim 18 further comprising the step of:
assigning a portion of the plurality of software-generated interrupt signals stored in the second storage device to represent interrupts from some interrupt sources generating hardware interrupts and having a corresponding interrupt prioritization level.

20. **(New Claim)** The method of claim 18 further comprising the step of:
assigning a portion of the plurality of software-generated interrupt signals stored in the second storage device to represent interrupts from some interrupt sources generating hardware interrupts and having an interrupt prioritization level which differs from the interrupt prioritization level of the plurality of hardware-generated interrupt sources coupled to the first storage device.

21. **(New Claim)** The method of claim 18 further comprising the step of:
changing prioritization level of a predetermined hardware-generated interrupt by providing a software-generated interrupt which represents a corresponding hardware-generated interrupt source for the predetermined hardware-generated interrupt but with a different prioritization level than the predetermined hardware-generated interrupt.

B1 Cont'd

22. **(New Claim)** A method for implementing interrupts in a data processing system, comprising the steps of:

providing a first storage device having a plurality of inputs, each of the plurality of inputs being coupled by a respective physical conductor to one of a plurality of hardware-generated interrupt sources which selectively generate hardware interrupts and selectively storing the hardware interrupts, the first storage device providing one or more hardware-generated interrupt signals;

providing a second storage device having one or more inputs, each of the one or more inputs receiving and storing a predetermined one of a plurality of software-generated interrupt signals, at least some of the predetermined plurality of software-generated interrupt signals indicating an interrupt from a different source or of a different type than the hardware interrupts, the second storage device providing one or more software-generated interrupt signals;

coupling logic circuitry to the first storage device and the second storage device for receiving the one or more hardware-generated interrupt signals and the one or more software-generated signals, the logic circuitry providing an interrupt request signal which will cause an interrupt to occur in the data processing system; and

changing prioritization level of a predetermined hardware-generated interrupt by providing a software-generated interrupt which represents a corresponding hardware-generated interrupt source for the predetermined hardware-generated interrupt but with a different prioritization level than the predetermined hardware-generated interrupt.